UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/930,325	08/15/2001	Ulises J. Cicciarelli	RSW920010066US1	3522

7590 08/26/2004		

IBM Corporation T81/062
PO Box 12195
Research Triangle Park, NC 27709

EXAMINER
VU, TUAN A

ART UNIT	PAPER NUMBER
2124	

DATE MAILED: 08/26/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on *15 August 2001*.

2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) *1-16* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☒ Claim(s) *1-16* is/are rejected.

7) ☐ Claim(s) _____ is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.

10) ☒ The drawing(s) filed on *15 August 2004* is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a) ☐ All   b) ☐ Some * c) ☐ None of:

      1. ☐ Certified copies of the priority documents have been received.

      2. ☐ Certified copies of the priority documents have been received in Application No. _____.

      3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *20010815, 20020912*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____ .

DETAILED ACTION

1.   This action is responsive to the application filed August 15, 2001.

Claims 1-16 have been submitted for examination.

Double Patenting

2.   The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969).
        A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).
        Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3.   Claims 1-16 are provisionally rejected under the judicially created doctrine of

obviousness-type double patenting as being unpatentable over claims 6, 12, 17 of copending

Application No. 09/930,359 (hereinafter '359), in view of Nabahi, USPN: 6,266,811 (hereinafter

Nabahi).

        Although the conflicting claims are not identical, they are not patentably distinct from

each other because of the following observations.

        Following are but a few examples as to how the certain claims from the instant invention

and from the above copending application are conflicting with each other.

        As per instant claim 1, '359 claim 6 also recites 'defining an object model representing a

plurality of components of a software installation package and one or more topology objects ...

and populating the object model to describe a particular software installation package and one or

more topologies for deployment of that ... package'; selecting one of the topologies and using it with a populated model for taking the action to install the particular software installation package. '359 claim 6 does not recite defining one or more rules for execution by a rules engine, wherein each rule specifies one or more conditions and at least one action to be taken when the conditions are matched, and the conditions pertain to a target run-time environment. But '359 recites identifying target machines on which the package is to be installed and performing the installation upon such identifying; thus has suggested mapping the runtime environment conditions against the specifications of the populated model and the topology selected, hence against some required specifications or rule-like conditions to meet. The use of requirements or rules in an installation process of installation package and mapping target environment with predefined and required conditions was disclosed by Nabahi, who discloses rule-based engine provided through script execution adapted to meet persisted set of rules according to requirements of target computer operating environment (e.g. col. 5; Fig. 1-4). Hence, the limitation as to use a rule-based installation process or script language enforcing as taught by Nabahi so that some conditions are to be met as suggested by '359 would have been obvious because rule and script-based can be structured and modified by developers enabling more flexible way to accommodate the installation to meet the runtime target environment in a more controllable manner.

As per instant claims 9, and 13, these are, respectively, system and product claims corresponding to instant claim 1 and map with '359 system and product claims 12 and 17, respectively; hence are rejected with the same rationale as set forth above.

Information Disclosure Statement

4.   The information disclosure statement (IDS) filed 8/2001 and 9/2002 fails to comply with

37 CFR 1.98(a)(3) because it does not include a concise explanation of the relevance in regard to

some content included, as it is presently understood by the individual designated in 37 CFR

1.56(c) most knowledgeable about the content of the information. It has been placed in the

application file and very superficially and reluctantly considered because it appears that the

subject matter presented therein exhibits no relevance whatsoever to the claimed invention.

In fact, regarding all the documents presented in the IDS, whether non-patent or patent

documents, the content or subject matter thereof is predominantly about features or technological

breakthroughs in some medical, pathological, or biological field or domain. It is urged that in

order for the document to be fully considered Applicant provide appropriate explanation as to

how the subject matter of said document relates to the present invention.

Claim Rejections - 35 USC § 103

5.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

6.   Claims 1-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Lister et al.,

USPN: 5,966,540 (hereinafter Lister), in view of Shing et al., USPN: 5,495,610 (hereinafter

Shing), and Charisius et al., USPubN: 2002/0104071 (hereinafter Charisius); and further in view

of Nabahi USPN: 6,266,811.

As per claim 1, Lister discloses a method of improving installation of software

packages, comprising:

defining an neutral object representing a plurality of components of a software

installation package (Java Installer – Fig. 1; col. 2, lines 22-44 – Note: platform independent

aspect of object being the Java installer is equivalent to a neutral object) and

one or more topology objects (e.g. set of operating system - col. 2, lines 44-62; *OS/2* box

- Fig. 3), wherein each component (e.g. F1, F2, F3 – Fig.3) comprises a plurality of objects (e.g.;

class Jinst – col. 6, line 52 to col. 7, line 34 – Note: Java implemented class objects compose the

component represented by F1…F3 functions of Fig. 3) and

each topology object identifies one or more selected one of the components (e.g. Fig. 3;

col. 3, lines 43-52 – Note: each identified operating system requiring a set of functions reads on

each topology selecting one or more components).

But Lister does not explicitly disclose that the neutral object is a model object; nor does

Lister disclose populating the object model to describe the installation package and the

topologies for deployment thereof.   However, Lister intends to develop code in neutral form to

accommodate for set of operating systems, some real-world requirements; putting in evidence

the import of the concept of object-oriented like Java in software development to address

heterogeneous platforms and environments or portability requirements. The use of object–based

design or requirement engineering using high-level abstraction supporting most OO software

development or framework was a known concept in the software engineering at the time the

invention was made. Based on the heterogeneous platforms accommodation via the installation

implementation by Lister, the development of the Java installation code at a higher level so to

include the object-oriented high-level abstraction before code implementation or specific runtime

or real-world instantiation is implicitly suggested. Shing, in a method to support installation

using analysis of target hardware/operating system analogous to Lister, discloses using modeling

(ERDs, CASE) in conjunction with a repository of rules during the building of the software

entities to stage installation commands at the target machines (e.g. Figs. 3-4) . Further, the use

of model to support such high-level requirement is exemplified by Charisius. Charisius, in a

method to provide Java-based deployment package (see para 0038, Fig. 38-39) to end users,

analogous to a language-neutral package installer by Lister, discloses the use of a model object

using a UML standard (e.g. Fig. 12-19) and Case tools to identify use cases or topology similar

to a operating system set or scenario by Lister. Hence, it would have been obvious for one of

ordinary skill in the art at the time the invention was made to develop the Java package as taught

by Lister using a object model, such model being populated with specification or description on

package feature for installation and deployment as taught in the case tool by Charisius or Shing

because the object model would identify the scenarios under which a set of functionalities need

to be identified for fulfilling the platform –independent requirements of the installation package

intended by Lister. The motivation would be that using a model as being a well-known feature

in the art of modeling with Case tool or Use Cases analysis such as exemplified by Charisius's

UML-based tool the requirements analysis or Shing's Case tool, validation or modification

would be enhanced whereby the dynamics of the real-world requirements would be laid out and

tested prior to deployment or even code implementation (see Charisius para 0034-0045); or that

relationships between hierarchical of real-world systems can be evidenced and rationally dealt

with (see Shing: Background of Invention).

Nor does Lister disclose defining one or more rules for execution by a rule engine, each

rule specifying one or more specified conditions are to be matched and an action to be taken

when the rule engine is executing, and wherein said conditions pertain to a target runtime

environment and at least one action may be used to select from among the topologies. Lister

discloses a script to execute some conditions matching (e.g. col. 7, lines 9-10), conditions to

meet when identifying a set of commands or functions that are matched with a particular

operating system (see Fig. 3); hence has suggested a rule specifying a specified set of conditions

and an action (or set of commands in F1...F3) for said conditions; and this matching using rules

is disclosed by Shing from above. Further, Nabahi, in a method to provide a installation package

analogous to Lister, discloses a rule-based installation engine executed via a script program (e.g.

Fig. 4). In view of the matching as suggested by Lister and approach by Shing, would have been

obvious for one of ordinary skill in the art at the time the invention was made to provide a rule-

based engine executed via a program with commands and syntax (see Nabahi: Summary of

Invention , table 1-4) because this would enable a solid form for matching the conditions of the

runtime installation with a modifiable set of rules defining the predetermined actions selected

from the requirements analysis as intended by Lister with modeling enhancement as taught by

Shing or Charisius.

As per claim 2, based on the model as taught by Charisius, the populating of instantiated

objects created from the Case Tool (Fig. 12-19) would have been obvious by virtue of the

rationale as set forth in claim 1.

As per claim 3, Charisius teaches JavaBeans (para 0019) and this would make this

limitation obvious because at the time the invention was made, Java programming has extended

into meeting complexity of distributed business transactions; and providing implementation in

JavaBeans would facilitate the management and rendering of enterprise applications in a dynamic manner as suggested via Charisius's approach for deployment (para 0019-0033).

As per claim 4, Lister in conjunction with the modeling by Charisius discloses instantiation into objects for a particular component identified for a particular operating system topology and one or more components objects for each component included in the particular package for that topology (refer objects, component, topology, and installation package from claim 1).

As per claim 5, Lister in conjunction with Nabahi discloses information about the target runtime environment; and using the information as input into the rule engine; selecting based upon the matching rule at least one topologies for deployment (Fig. 3 – Note: runtime platform information and mapping conditions derived from such information and selecting a set of functions from such rule like mapping is equivalent to discovering, using it for input into the rule engine, and selecting the topologies for deployment based thereupon); but Lister does not teach dynamically discovering information pertaining to the runtime environment for feeding the rule engine, from there, feeding into the rule engine, selecting a topology and using the populated the object model based on the selected topology. In view of the use of model and Javabeans as taught by Charisius, the dynamic limitation as in discovering of information and feeding into the rule engine as well as selecting a topology and using populated model for that selection would have been obvious for the same rationale as in claim 1; and also of an implicit teaching of such limitation by Charisius. When the model is being populated and the Case tool is executing by Shing/Charisius, those above steps are dynamic to that runtime of the tool; and the selected

model instantiated from a selected topology based on runtime identifying of target environment as well as its mapping into the rule engine would have been the dynamic result of those steps.

As per claim 6, Lister discloses identifying target machines on which to apply the installation, downloading to the identified target and performing the installation using such installation package (see col. 7, line 35 to col. 8, line 17); the step of using the populated object model being addressed in claim 5.

As per claim 7, Lister discloses multi-platform or operating system-neutral installation package (re claim 1) and communication scheme (Fig. 1A); but does not teach authentication of server operating the downloading step. Official notice is taken that the concept of identifying who is the source of the transmitted data over a network, i.e. authentication process, was a known concept in the art of software distribution in corporate and internet transactions as suggested by Charisius's support of multi-tiered corporate transactions (see Charisius: Background of Invention), at the time the invention was made. Adding this authentication process to the installation process by Lister would have been obvious because it would not benefit the recipient of the installation package should the source provider turns out to be an untrusted and harmful source and that the data being installed would cause harm to the system target where the data has been installed.

As per claim 8, Lister discloses information to the rule engine serves to configure values needed by the topology (e.g. *OSList = { NONE, OS/2 ...etc }*; *set our OS variable* – col. 6, line 53 to col. 7, line 8 – Note: if topology is operating system then variable being set according to the topology is in the script for mapping information and configure variable).

As per claim 9, this claim is the system claim with means to perform the exact steps corresponding to those of method claim 1; hence is rejected with the corresponding rejection as set forth therein.

As per claims 10 and 11, these claims correspond to claims 5 and 6, respectively, hence are rejected with the corresponding rejection as set forth therein.

As per claim 12, this claim corresponds to claim 8, and is rejected using the rejection therein.

As per claim 13, this claim is the computer-readable product claim with means to perform the exact steps corresponding to those of method claim 1; hence is rejected with the corresponding rejection as set forth therein.

As per claims 14 and 15, these claims are computer-readable product claims corresponding to claims 5 and 6, respectively, hence are rejected with the corresponding rejections as set forth therein.

As per claim 16, this claim is the computer-readable product claim corresponding to claim 8; hence is rejected with the corresponding rejection as set forth therein.

Conclusion

7.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Pub No. 2003/0028869 to Drake et al., disclosing OM modeling and JavaBeans in framework for implementing installation package.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (703)305-7207. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Kakali Chaki can be reached on (703)305-9662.

Any response to this action should be mailed to:

Commissioner of Patents and Trademarks

Washington, D.C. 20231

or faxed to:

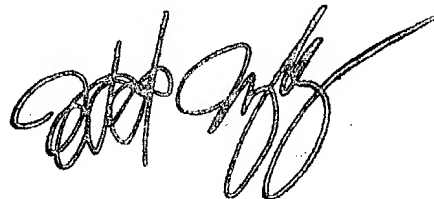(703) 872-9306 (for formal communications intended for entry)

or: (703) 746-8734 (for informal or draft communications, please label

"PROPOSED" or "DRAFT" – please consult Examiner before use)

Hand-delivered responses should be brought to Crystal Park II, 2121 Crystal Drive,

Arlington. VA. , 22202. 4th Floor(Receptionist).

Any inquiry of a general nature or relating to the status of this application or proceeding

should be directed to the receptionist whose telephone number is (703) 305-3900.

VAT
August 16, 2004

TODD INGBERG
PRIMARY EXAMINER